JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 28 No. 8, pp. 49-58, August 2023
https://doi.org/10.9708/jksci.2023.28.08.049

Performance Comparison of Neural Network and Gradient Boosting Machine for Dropout Prediction of University Students

Hyeon Gyu Kim*

*Associate Professor, Div. of Computer Science and Engineering, Sahmyook University, Seoul, Korea

[Abstract]

Dropouts of students not only cause financial loss to the university, but also have negative impacts on individual students and society together. To resolve this issue, various studies have been conducted to predict student dropout using machine learning. This paper presents a model implemented using DNN (Deep Neural Network) and LGBM (Light Gradient Boosting Machine) to predict dropout of university students and compares their performance. The academic record and grade data collected from 20,050 students at A University, a small and medium-sized 4-year university in Seoul, were used for learning. Among the 140 attributes of the collected data, only the attributes with a correlation coefficient of 0.1 or higher with the attribute indicating dropout were extracted and used for learning. As learning algorithms, DNN (Deep Neural Network) and LightGBM (Light Gradient Boosting Machine) were used. Our experimental results showed that the F1-scores of DNN and LGBM were 0.798 and 0.826, respectively, indicating that LGBM provided 2.5% better prediction performance than DNN.

▸ **Key words**: Dropout prediction, DNN, LGBM, Oversampling, SMOTE

[요 약]

학생들의 중도 탈락은 대학의 재정적 손실 뿐 아니라, 학생 개개인 및 사회적으로도 부정적인 영향을 끼친다. 이러한 문제를 해결하기 위해 기계 학습을 이용하여 대학생들의 중도 탈락 여부를 예측하고자 하는 다양한 시도가 이루어지고 있다. 본 논문에서는 대학생들의 중도 탈락 여부를 예측하기 위해 DNN(Deep Neural Network)과 LGBM(Light Gradient Boosting Machine)을 이용한 모델을 구현하고 성능을 비교하였다. 학습 데이터로는 서울 소재 중소규모 4년제 대학인 A 대학의 20,050명의 학생을 대상으로 수집된 학적 및 성적 데이터를 학습에 이용하였다. 원본 데이터의 140여개의 속성 중 중도 탈락 여부를 나타내는 속성과의 상관계수가 0.1 이상인 속성들만 추출하여 학습하였다. 두 모델의 성능 실험 결과, DNN과 LGBM의 F1-스코어는 0.798과 0.826이었으며, LGBM이 DNN에 비해 2.5% 나은 예측 성능을 보였다.

▸ **주제어**: 중도 탈락 예측, DNN, LGBM, 오버샘플링, SMOTE

• First Author: Hyeon Gyu Kim, Corresponding Author: Hyeon Gyu Kim
*Hyeon Gyu Kim (hgkim@syu.ac.kr), Div. of Computer Science and Engineering, Sahmyook University
• Received: 2023. 07. 21, Revised: 2023. 08. 21, Accepted: 2023. 08. 21.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

중도 탈락은 학생 개인뿐만 아니라 사회적으로도 부정적인 영향을 끼친다. 학생 개인의 측면에서는 학위 취득 상실로 인해 양질의 직업을 획득할 수 있는 기회가 제한되어 소득이 낮아질 수 있으며, 재입학이나 타 대학으로 편입하는 경우 부가적인 시간과 비용이 소모될 수 있다[1]. 또한 중도 탈락 학생들의 경우 스킬이 부족하여 별도의 취업 교육을 필요로 하거나 미취업 상태로 남아있을 수 있으며, 사회적 측면에서 재취업 교육비 및 실직 급여 등 사회 보장 비용의 증가 요인이 될 수 있다[2].

대학 측면에서도 학생의 중도 탈락은 직접적인 재정적 손실로 이어지며, 추가 학생 선발로 인한 비용 증가 등 재정 운영에 부정적인 영향을 미치게 된다. 2018년 기준, 4년제 고등교육 기관의 중도 탈락률은 약 6.6%이며, 사이버 대학의 경우 18.9%에 이르고 있다[3]. 따라서 다수의 대학에서 학생들의 중도 탈락 여부를 예측하기 위한 솔루션 개발에 많은 노력을 기울이고 있다.

중도 탈락률 예측과 개선을 위한 기존 연구는 크게 통계적 접근 방법과 기계 학습을 이용한 접근 방법으로 나뉠 수 있다. 통계적 접근 방법은 기존 데이터의 특성과 인과관계를 설명하는데 중점을 두는 반면, 새로운 데이터에 대한 예측력이 보장되지 않는 한계를 지닌다[4]. 이에 반해 기계 학습을 이용한 방법에서는 예측은 가능한 반면, 예측 결과에 대한 설명력이 다소 부족한 측면이 있다. 두 접근 방법 중 최근에는 기계 학습을 이용하여 예측 모델을 개발하고자 하는 시도가 더욱 활발히 이루어지고 있다.

기계 학습을 이용할 경우 중도 탈락률 예측의 정확도를 높이기 위해서는 데이터 특성을 알고리즘에 적절히 반영하는 것이 중요하다. 앞서 언급한 바와 같이, 4년제 대학의 경우 중도 탈락률이 약 6.6%이며, 중도 탈락 여부를 기준으로 데이터를 두 개의 클래스로 나눌 경우 클래스 간의 불균형이 심각한 구조를 지닌다. 따라서 데이터 불균형을 고려한 학습 방법을 채택해야 하며, 예측 성능을 평가하고자 할 경우 역시 불균형을 고려한 성능 지표를 채택하여 알고리즘을 비교할 필요가 있다.

본 논문에서는 서울 내 중소규모 4년제 대학인 A 대학의 사례를 중심으로 학생들의 중도 탈락 여부를 예측하기 위한 지도 학습 모델을 구축하고 성능을 비교하였다. 학습 데이터로는 A 대학의 학사 정보 시스템에서 2013년부터 2022년까지 20,050명의 학생을 대상으로 수집된 학적 및 성적 변동 데이터를 학습에 이용하였다. 원본 데이터의 140여개의 속성 중 중도 탈락 여부를 나타내는 속성과의 상관계수(Correlation Coefficient)가 0.01 이상인 속성들만을 추출하여 학습에 이용하였다. 학습 알고리즘으로는 심층 신경망(DNN, Deep Neural Network)[5]과 LGBM (Light Gradient Boosting Machine)[6]을 이용하였다. 그리고 데이터 불균형이 예측 성능에 미치는 영향을 확인하기 위해 SMOTE(Synthetic Minority Over sampling Technique)[7] 기법을 적용하여 데이터를 오버샘플링한 경우와 그렇지 않은 경우의 예측 성능을 비교하였다. 성능 지표로는 데이터 불균형을 고려한 F1-스코어를 채택하였다.

본 논문의 구성은 다음과 같다. 2장에서는 기계 학습을 이용한 중도 탈락률 예측과 관련한 기존 연구들을 소개한다. 3장에서는 학습에 이용된 데이터의 구조와 특성 및 전처리 과정 등에 대해 논의한다. 4장에서는 중도 탈락 예측을 위한 DNN과 LGBM의 모델 구성을 설명하고, 실험을 통해 얻어진 성능 결과를 비교한다. 그리고 오버샘플링이 예측 성능에 미치는 영향에 대해서도 함께 논의한다. 5장에서는 결론 및 추후 연구 방향 제시로 마무리한다.

II. Related Work

앞서 언급한 바와 같이, 중도 탈락률 예측과 개선을 위한 기존 연구는 통계적 접근 방법과 기계 학습을 이용한 접근 방법으로 나뉠 수 있다. 기존 연구의 예측 목적과 대상, 학습 데이터, 기계학습 알고리즘 등에 대해 Table 1에서 요약하였다.

먼저 통계적 접근 방법으로 [3]은 사이버 대학의 LMS 로그와 재학생 설문 조사 등의 자료를 바탕으로 통계적 방법을 이용한 중도 탈락 개선 모형을 구현하였으며, 모형의 실제 적용을 통해 신입생의 중도 탈락률 및 학업지속 비율을 향상시켰다. [8]은 성적, 결석, LMS 접속 횟수, 상담 횟수 등으로부터 선형회귀를 이용하여 성적을 예측하고, 이를 통해 대학생들의 학사경고 예측 요인을 분석하고자 하였다. [3, 8]에서는 공통적으로 통계적 방법을 활용한 현상 분석에 중점을 두었으며, 예측 성능과 관련된 내용은 언급되지 않았다.

최근에는 기계 학습을 이용하여 학생들의 성적이나 중도 탈락 여부를 예측하고자 하는 시도가 활발히 이루어졌다. 성적 예측과 관련하여, [4]는 26,000여 명의 대학생들의 LMS(Lecture Management System) 로그를 기반으로 학생 집단을 우수, 보통, 저성취 집단으로 나누고 학습을 수행하였으며, F1-score를 기준으로 우수 집단은 0.7, 나머지 집단은 0.2~0.3 정도의 예측 결과를 얻었다. [9]는

Table 1. Summarization of existing work: prediction purpose, target students, data, algorithms, etc.
(DT: Decision Tree, RF: Random Forest, DNN: Deep Neural Network, K-NN: K-Nearest Neighbor, SVM: Support Vector Machine)

	Ref#	Prediction purpose	Target students	Data	Algorithms	Best Prediction Result
Statistical approach	[3]	Dropout	Cyber university	2017~2018 LMS log + Survey from 1050 students	DT + Statistics	-
	[8]	Grade	University	2019 LMS log and academic DB	Linear Regression + Statistics	-
Machine learning approach	[4]	Grade	University	2020-1 LMS log (from 26,331 students)	DT, RF, DNN	F1-score: 0.7 (DNN) (for good performer group)
	[9]	Grade	Cyber university	2015~2017 academic DB from 1,113 students	AutoEncoder + DNN	Accuracy: 0.86 (DNN)
	[10]	Grade	Middle school	2019 academic DB from 2,500 students	DT, RF, K-NN, SVM, XGBoost	F1-score: 0.81 (XGBoost)
	[11]	Grade	Middle school	UCI Machine Learning Repository	DT, RF, XGBoost, LGBM	AUC: 0.803 (LGBM)
	[13]	Dropout	University	2011~2014 academic DB from 8,000 students	DT, DNN	Accuracy: 0.8 (DNN)
	[14]	Dropout	University	2017~2021 survey from 3,075 students	DT, RF, Naive Bayes, Ridge Regression	Precision: 0.739 (Ridge Regression)
	[15]	Dropout	University	-	DNN, Linear Regression	F1-score: 0.8 (DNN)
	[16]	Dropout	Cyber university	2012~2019 LMS log (from 98,685 students)	DT, RF, DNN, SVM	F1-score: 0.84 (RF)
	[17]	Dropout	MOOC	2014 LMS log from 3,617 students	DT, DNN, K-NN, SVM	AUC: 0.968 (DNN)
	[18]	Dropout	High school	2014 NEIS DB from 165,715 students	RF, Boosted DT	AUC: 0.898 (Boosted DT)

사이버대학 학생들을 대상으로 오토인코더로 출결을 예측한 후 DNN을 이용해 성적을 예측하였으며, 예측 정확도(Accuracy)는 86%로 조사되었다. [10]은 부산 지역의 중학생을 대상으로 K-NN(K-Nearest Neighbor), SVM, DT (Decision Tree), RF(Random Forest), XGBoost 등을 이용하여 성적을 예측하고자 하였으며, XGBoost의 정확도가 평균 81% 수준으로 가장 높은 것으로 조사되었다. [11]은 포르투갈 Alentejo 지역의 중학생을 대상으로 DT, RF, XGBoost, LGBM 등을 이용하여 성적을 예측하고자 하였으며, LGBM의 AUC(Area Under the ROC Curve) [12] 값이 0.803으로 가장 높았다.

중도 탈락 예측과 관련하여, [13]은 DT와 DNN을 이용하여 대학생들의 졸업 여부를 예측하고자 하였으며, 각각의 정확도는 75%와 80% 수준으로 DNN의 정확도가 우수하였다. [14]는 중도 탈락자 및 재학생 대상으로 수행된 설문 조사를 이용하여 대학생들의 중도 탈락 여부를 예측하고자 하였다. 텍스트 처리를 위해 Twitter 형태소 분석기를 이용하여 명사만을 추출하여 학습 데이터를 구축한 후, 나이브 베이즈(Naive Bayes), 릿지 회귀(Ridge Regression), DT, RF 등을 이용하여 예측 모형을 구현하였으며, 릿지 회귀의 정밀도(Precision) 값이 73.9%로 가장 높았다. [15]는 로지스틱 회귀와 DNN을 이용하여 대학생들의 중도 탈락을 예측하고자 하였으며, DNN의 F1-스코어가 0.8로 로지스틱 회귀에 비해 우수한 것으로 조사되었다.

[16]은 사이버 대학을 대상으로 한 중도 탈락 예측 문제를 논의하였다. LMS 로그를 이용하여 학습 데이터를 구축한 후 DT, RF, SVM, DNN의 예측 모형을 구현하였으며, RF의 F1-스코어가 0.84로 가장 높았다. [17]은 MOOC 과정 수강생을 대상으로 K-NN, SVM, DT, DNN의 중도 탈락 예측 정확도를 비교하였으며, DNN의 AUC 평균값이 0.968로 가장 높게 나타났다. [18]은 고교생을 대상으로 중도 탈락 예측 모형을 구현하였으며, Boosted DT의 AUC 평균값이 0.898로 가장 높았다.

제안 방법은 4년제 대학교 학생들의 중도 탈락 예측 문제를 다룬다는 점에서 [13-15]와 유사하다. 단, [13]과 [14]에서는 예측 성능을 측정하기 위한 지표로써 각각 정확도(Accuracy)와 정밀도(Precision)을 이용하였으며, 이들 척도는 데이터가 불균형할 때 이용되기에는 한계가 있다. 예를 들어, 중도 탈락률이 6.6%인 데이터에서 어떤 예측 모형이 데이터의 모든 레코드에 대해 중도 탈락을 하지 않는다고 단순하게 예측할 경우, 해당 모형의 정확도는 93.7%가 된다. 이와 같이 정확도는 중도 탈락으로 예측했으나 실제 하지 않은 경우나 그 반대의 경우에 대한 예측 오류들을 충분히 반영하지 못한다는 한계가 있다. 정밀도 역시 동일한 맥락에서 한계가 있다. [15]의 경우 F1-스코어를 이용하여 데이터 불균형을 반영한 정확도를 측정하

였다는 점에서는 제안 방법과 유사하나, 학습을 위한 데이터 종류, 전처리 방법, 학습 알고리즘의 하이퍼 파라미터 설정 방법 등에 대한 구체적인 구현 내용을 제시하지 않았다는 점에서 한계가 있다. IV장에서는 실험을 통해 제안 방법과 [13-15]와의 성능을 비교 결과를 제시한다.

III. Data Preparation

1. Data Description

중도 탈락 여부를 예측하기 위한 학습 데이터로 서울 내 중소규모 4년제 대학인 A 대학의 학사 정보 시스템에서 2010년부터 2022년까지 20,050명의 학생을 대상으로 수집된 학적 및 성적 변동 데이터를 학습에 이용하였다. 학사 정보 시스템의 원본 데이터는 데이터베이스 정규화를 통해 중복을 최소화할 수 있도록 여러 개의 테이블로 분리된 형태를 지닌다. 이 중 중도 탈락 예측에 이용될 수 있는 주요 테이블과 속성(컬럼)들은 Fig. 1과 같다. 해당 테이블들은 순서대로 성적, 학적 상태, 장학금, 상담 횟수, 비교과 참여, 도서 대출, 학생 정보 관련 데이터를 포함한다. 지면 관계상 각 속성에 대한 설명은 생략한다.

Table: Grade (with 28 columns)

SID	Year	Semester	Grade	NumCourse	NumF	...

Table: AcademicStatus (with 8 columns)

SID	Year	Semester	Status	Desc	Transfer	...

Table: Scholarship (with 8 columns)

SID	Year	Semester	Tuition	Scholarship	Admis	...

Table: Counsel (with 4 columns)

SID	Year	Semester	NumCounsel

Table: ExtraCurricular (with 16 columns)

SID	Year	Semester	NumExtra	NumVolunteer	...

Table: BookLoan (with 10 columns)

SID	Year	Semester	NumLoan	AvgDuration	...

Table: StudentInfo (with 70 columns)

SID	Name	Dept	Major	AdmisYear	AdmisType	...

Fig. 1. Structures of major tables that can be used for dropout prediction

Fig. 1에서 밑줄 친 속성은 기본키(Primary key)로 이용되는 속성을 의미한다. StudentInfo를 제외한 모든 테이블은 학번(SID)과 기준연도(Year), 학기(Semester)를 조합한 형태의 기본키를 이용한다. 신입학 학생이 휴학 없이 졸업할 경우 8학기를 이수하게 되므로, 일반적인 경우 학생별로 8개의 레코드가 포함된다. 여기에 중도 탈락이나 휴학 등의 사유로 인해 학생별 레코드 수는 가변적이며, 최소 1개에서 최대 19개까지의 레코드를 가지는 것으로 조사되었다. Fig. 2는 Grade 테이블에서 정상적으로 졸업한 학생(2012xxx010)과 중도 탈락한 학생(2012xxx011)의 레코드 예를 보여준다.

Table: Grade (with 28 columns)

SID	Year	Semester	Grade	NumCourse	...
2012xxx010	2012	1	3.53	17	...
2012xxx010	2012	2	2.69	16	...
2012xxx010	2013	1	3.17	18	...
2012xxx010	2013	2	2.53	16	...
2012xxx010	2014	1	2.93	17	...
2012xxx010	2014	2	4.07	17	...
2012xxx010	2015	1	3.78	19	...
2012xxx010	2015	2	3.72	17	...
2012xxx011	2012	1	0.45	16	...
2012xxx012	2012	1	4.12	15	...
...	...	...	...	...	...

Fig. 2. Record examples of graduate and dropout students in the Grade table

Grade, AcademicStatus 등 StudentInfo를 제외한 모든 테이블은 학기별로 학생들의 정보를 저장하고 있으며, 이들 테이블의 레코드 수는 168,000여 개로 동일하였다. StudentInfo는 20,050명의 학생을 대상으로 SID 별로 학생들의 이름과 학과, 전공, 입학년도, 입학 형태(신입학, 편입학 등) 등의 정보를 포함한다.

2. Feature Selection

제안 방법에서는 학습에 필요한 데이터를 생성하기 위해 SID 값을 기준으로 Fig. 1의 테이블들을 하나로 병합하였다. 병합 시 고려해야 할 사항은 다음과 같다.

- StudentInfo를 제외한 모든 테이블은 학기별로 정보를 포함한다. 따라서 이들 테이블에서 같은 SID 값을 가지는 여러 개의 레코드를 요약하여 하나의 레코드로 변환해야 한다.
- 7개의 테이블을 모두 병합할 경우, 144개의 속성이 테이블에 추가된다. 이들 중 모든 속성이 중도 탈락에 밀접한 영향을 주는 것은 아니다. 연관성이 적은 속성을 추가하여 학습할 경우 예측 성능이 떨어질 수 있다. 따라서 높은 예측 성능을 얻기 위해서는 중도 탈락과 높은 상관관계(Correlation)를 지니는 특징(Feature) 속성들만을 추출하여 학습할 필요가 있다.

첫 번째 사항과 관련하여, 테이블 별로 요약 정보를 추출하는 방법은 다음과 같다. 먼저 Grade 테이블에서는 동일한 SID 값을 지니는 N개의 레코드(학기별 성적 정보)로부터 최근 학기의 성적에 높은 가중치를 주어 합산한 값을 요약 레코드로 추출한다. 가중치 값으로는 단순 지수평활(Simple exponential smoothing)[19] 함수를 이용하였다. 학적 상태를 나타내는 AcademicStatus 테이블에서는 학기별 학적 상태를 나타내는 Status 속성값을 이용해 졸업이나 중도 탈락 직전의 연속 휴학 학기 수를 계산하여 요약 레코드로 추출한다. Status 속성에는 "신입학", "편입학", "재학", "진급", "일반휴학", "군휴학", "졸업", "제적" 등의 학기별 학적 상태를 나타내는 값이 포함되어 있으며, 속성값이 "휴학"으로 끝나는 학기의 수를 카운트한다. 단, "군휴학"일 경우에는 카운트에서 제외한다. 이외에 장학금, 상담 수, 비교과 참여 여부, 도서 대출 등에 데이터에 대해서도 유사한 카운팅 방법을 적용하여 SID 별로 요약 레코드를 추출한다.

Table 2. Attributes of a merged table to perform a machine learning

Column	Description	Source table
SID	Student ID (primary key)	All
Grade	Average grade (higher weight for recent values)	Grade
NumAbs	Number of consecutive semesters of absence right before graduation or dropout	Academic Status
Scholar	Cumulative scholarship amount right before graduation or dropout	Scholar ship
NumCouns	Cumulative number of counseling right before graduation or dropout	Counsel
NumExtra	Cumulative number of participation in extra curriculum subjects right before graduation or dropout	Extra Curricular
NumBook	Cumulative number of bool loans right before graduation or dropout	BookLoan
NumSem	Number of semesters excluding semesters of absence	Grade
NumF	Cumulative number of "F" grades right before graduation or dropout	Grade
Dept	Department of the student with SID	Student Info
AdmType	Admission type of the student with SID	Student Info
...	...	...
Dropout	Final status: dropout or not (Required for the supervised learning)	Academic Status

이외에도 Fig. 1의 테이블로부터 학습 정확도를 높이기 위한 부가적인 속성들을 추출하여 학습에 이용할 수 있다. 예를 들어, Grade 테이블에서는 성적 외에 학생 별로 재학 학기 수나 F 학점 수 등을 추출할 수 있다. 재학 학기의 경우, 동일한 SID 값을 지니는 레코드로부터 휴학 학기를 제외한 레코드 수를 합산하여 구할 수 있다.

각 테이블에서 SID 별로 요약 레코드를 얻고 나면, SID 값을 기준으로 하나의 테이블로 병합한다. 그리고 지도 학습 수행을 위해 각각의 레코드에는 중도 탈락 여부를 나타내는 Dropout 속성을 추가한다. 이는 AcademicStatus 테이블의 Status 속성값을 이용하여 구현할 수 있으며, 속성값이 "제적"으로 끝나는 경우 Dropout 속성값을 1로, 이외의 경우에는 0으로 설정한다. 이외에 StudentInfo 테이블로부터 학과 및 입학 형태 등 학습에 도움을 줄 것으로 예상되는 속성들을 병합 테이블에 추가한다. Table 2는 학습을 위해 생성된 병합 테이블에 포함된 속성과 내용, 속성값을 가져온 원본 테이블 등에 대해 보여준다. 병합 테이블의 기본 키는 SID가 된다.

병합 테이블에는 Fig. 1로부터 산술적으로 최대 140여 개의 속성이 포함될 수 있다. 앞서 언급한 바와 같이 예측 성능을 높이기 위해서는 Dropout 속성과 상관관계가 높은 속성만을 추출하여 학습 데이터를 구성할 필요가 있다. Fig. 3은 병합 테이블의 속성 중 Dropout과의 상관계수 값이 0.01 이상인 속성들을 보여준다.

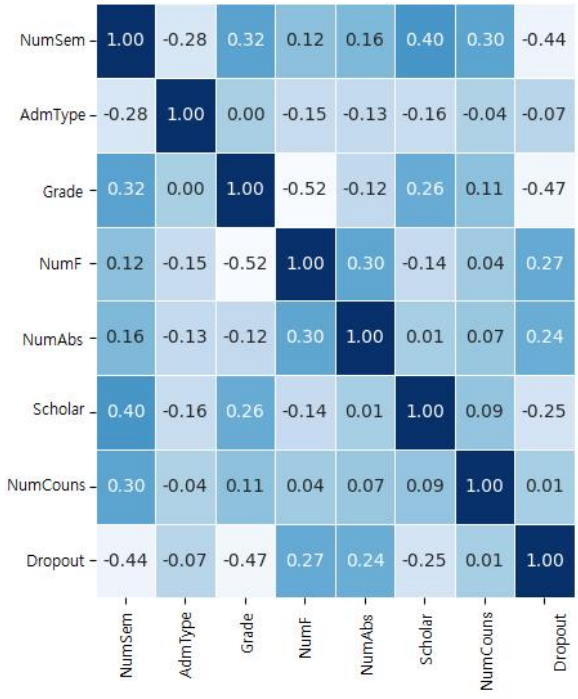

Fig. 3. Attributes of the merged table with a correlation coefficient ≥ 0.01 with the Dropout attribute

Fig. 3은 Dropout과 상관관계가 가장 높게 나타난 속성은 성적이었으며, 다음으로 재학 학기 수, F-학점 수, 장학금, 휴학 학기 수 등의 순이었다. 재학 학기 수의 경우, 성적 및 장학금과 상관관계가 높게 나타났으며, 성적은 F-학점 수와 높은 상관관계를 보였다. 제안 방법에서 속성 간의 상관계수를 구하기 위해, 파이썬의 데이터 프레임이 제공하는 corr() 함수를 이용하였으며, 시각화를 위해 Seaborn 패키지의 heatmap() 함수가 이용되었다.

제안 방법에서는 Dropout 속성과의 상관계수가 0.01 이상인지 여부를 선택 기준으로 활용하였으며, 결과적으로 Fig. 3에서 제시한 8개의 속성을 학습을 위한 입력 값으로 이용하였다.

IV. Dropout Prediction

1. Considering Data Skewness

중도 탈락 예측과 같은 2-클래스 분류 문제에서 예측 성능을 판단하기 위한 경우의 수는 4 가지로써 TP(True Positive), FP(False Positive), FN(False Negative), TN(True Negative)이 존재한다. True는 예측이 적중한 경우를, False는 그렇지 않은 경우를 의미한다. Positive는 중도 탈락하는 경우를, Negative는 그렇지 않은 경우를 의미한다. 따라서 TP는 중도 탈락으로 예측한 것이 맞았을 경우를, FP는 중도 탈락으로 예측하였으나 예측이 틀린 경우를 의미한다. FN과 TN도 동일한 방법으로 해석 가능하다.

정확도(Accuracy)는 예측 성능을 판단하기 위한 가장 대중적인 성능 지표로써 아래와 같이 정의된다.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

단, 데이터가 한쪽 클래스로 편중될 경우, 정확도로 성능을 측정하는 방법에는 한계가 있다. 예를 들어, 1장에서 언급한 바와 같이, 4년제 대학의 중도 탈락 비율은 평균 6.6%로 알려져 있다. 이 경우 높은 정확도를 얻을 수 있는 방법 중 하나는 모든 경우에 대해 단순히 중도 탈락하지 않는 것으로 예측하는 것이다. 이 경우 정확도는 93.4%가 된다. 반대의 경우 정확도는 6.6%로 현저히 낮아진다.

위 문제는 FP와 FN 관점에서 각각 오류율을 측정하고 성능에 적절히 반영해야 함을 보여준다. 정밀도(Precision)는 FP의 관점에서 성능을 측정하기 위한 지표로써 아래와 같이 정의된다.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

재현율(Recall)은 FN 관점에서 성능을 측정하기 위한 지표로써 아래와 같이 정의된다.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

FP와 FN 모두를 고려한 정확도를 측정하기 위한 간단한 방법은 정밀도와 재현율의 평균값을 이용하는 것이다. F1-스코어는 정밀도와 재현율의 조화 평균으로써 아래와 같이 정의된다.

$$F1-score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

결과적으로 데이터에 불균형이 있을 경우 예측 성능을 측정하기 위해서는 F1-스코어를 이용하는 것이 바람직하다. 제안 방법 역시 중도 탈락 예측 성능 측정을 위해 F1-스코어를 이용한다. 논의의 편의를 위해, 중도 탈락 데이터를 모은 클래스를 P(Positive), 중도 탈락이 아닌 데이터를 모은 클래스를 N(Negative)로 명명한다.

한편 데이터가 한쪽 클래스로 편중될 경우, 학습 시 분포도가 높은 클래스에 많은 가중치가 부여되므로 과적합(Overfitting) 문제가 발생할 수 있다. 이 경우 분포도가 높은 클래스로 학습이 편중되며, 결과적으로 N 클래스의 예측 성능은 높아지는 반면, P 클래스의 성능은 낮아지는 문제점이 발생할 수 있다.

데이터 불균형으로 인한 과적합 문제를 해결하기 위해 가장 대중적으로 이용되는 방법은 SMOTE 기법을 이용한 오버샘플링이다. SMOTE는 K-NN(K-Nearest Neighbor) 기법을 기반으로 소수 클래스의 데이터를 임의로 증식한다. 중도 탈락 예측 문제에 SMOTE를 적용할 경우, 먼저 소수 클래스인 P 클래스의 데이터를 추출한 후, K-NN을 이용해 특정 기준 벡터와 가까운 K개의 이웃 벡터를 선정한다. 그리고 기준 벡터와 이웃 벡터들을 선으로 연결한 후, 연결선 사이의 임의의 점들을 추출하여 학습 데이터에 추가시켜 데이터를 증식한다.

주의할 점은 오버샘플링이 반드시 더 나은 예측 성능으로 이어지지는 않는다는 점이다. 예를 들어, 증식된 P 클래스의 데이터가 기존의 N 클래스의 데이터 영역과 겹칠 경우 노이즈로 작용되어 오히려 예측 성능을 떨어뜨릴 수 있다. 이와 관련하여 IV.3절에서는 실험을 통해 SMOTE 적용 전과 후의 성능을 제시하고 결과에 대해 함께 논의한다.

2. Algorithm Settings

제안 방법에서는 중도 탈락 예측을 위한 학습 알고리즘으로 심층신경망(DNN)과 LightGBM(LGBM)을 이용하였다. DNN은 가장 기본적인 신경망 알고리즘이며, LGBM은 결정 트리(Decision Tree)를 앙상블한 형태의 알고리즘 중 하나로써 비교적 적은 비용으로 높은 정확도를 얻을 수 있어 최근 다양한 응용에서 활용되고 있다. 학습 모델의 구현에는 구글에서 제공하는 인공 신경망 오픈 소스 소프트웨어 라이브러리인 Keras[20]와 Python 언어가 이용되었다.

먼저 입력 데이터 및 학습 데이터 구성은 다음과 같다. 각각의 데이터는 데이터 프레임 형태로 저장되며, X는 입력 레코드, T는 레코드별 중도탈락 여부를 나타낸다. 입력 레코드에서 각 속성의 수치값에는 편차가 있으므로, 사이킷런(Scikit-learn) 모듈의 StandardScaler를 이용하여 0과 1 사이의 값으로 모두 정규화(Normalize)한다. 그리고 사이킷런의 train_test_split() 함수를 이용하여 학습 데이터 셋과 테스트 데이터 셋을 8:2의 비율로 분리한다.

```
X = df[["AdmType","NumSem","Grade", ... ]].values
T = df["Dropout"].values

X_scaled = StandardScaler().fit_transform(X)
X_train, X_test, T_train, T_test =
    train_test_split(X_scaled, T, test_size=0.2)
```

위 학습 데이터로부터 중도 탈락 여부를 예측하기 위한 DNN 모델의 구성은 다음과 같다. 모델은 크게 4개의 Dense 층으로 구성되며, 각각의 Dense 층은 히든 노드의 개수와 활성화 함수를 지정하기 위한 2개의 파라미터를 지닌다. 히든 노드는 학습 정보에 해당하는 가중치(Weight)와 편향(Bias) 정보를 가지며, 활성화 함수는 다음 층으로 내보낼 값을 계산하기 위한 함수를 지정한다. 마지막 층을 제외하고는 모두 Relu 함수로 지정함으로써 층의 개수가 많아지더라도 기울기 소실(Vanishing Gradient) 문제가 발생하지 않도록 구성하였다. 이외에 첫 번째 Dense 층에는 입력 데이터의 크기를 지정하기 위한 input_shape 속성을 부가적으로 정의하였다. 그리고 마지막 Dense 층에서는 sigmoid 함수를 이용해 중도 탈락 예측값이 0과 1사이의 확률값이 출력될 수 있도록 구성하였다. Dense 층 사이에는 Dropout 층을 추가해 과적합을 방지하고자 하였다.

```
model = keras.Sequential([
    keras.layers.Dense(256, activation="relu",
        input_shape=(X,shape[1], )),
    keras.layers.Dropout(rate=0.5),
    keras.layers.Dense(128, activation="relu"),
    keras.layers.Dropout(rate=0.5),
    keras.layers.Dense(32, activation="relu"),
    keras.layers.Dense(1, activation='sigmoid')
])
model.compile(optimizer="adam", loss='binary_cross...')
model.fit(X_train, T_train, epochs=100, callback=...)
```

DNN 모델의 옵티마이저로는 adam을 채택하였으며, 가중치와 편향 값을 보정하기 위한 손실 함수로는 이진 분류에 활용되는 binary_crossentropy를 정의하였다. 그리고 X_train과 T_train을 훈련 데이터로 입력받아 학습을 수행할 수 있도록 fit() 함수를 정의하였다. 학습은 최대 100회까지 수행될 수 있으며, 손실 함수 값이 일정 크기 이상 감소하지 않을 경우 학습은 중단될 수 있도록 콜백 함수를 fit()에 추가하였다. (지면 관계상 콜백 함수 정의 부분은 생략하였다.)

다음으로 중도 탈락 예측을 위한 LGBM 모델의 구성은 다음과 같다. LGBM의 경우 실행을 위한 다양한 하이퍼 파라미터 설정이 필요하다.

```
import lightgbm as lgb

D_train = lgb.Dataset(X_train, label=T_train)

params = {}
params['learning_rate'] = 0.003
params['boosting_type'] = 'gbdt'
params['objective'] = 'binary'
params['metric'] = 'binary_logloss'
params['num_leaves'] = 500
params['min_data'] = 100
params['max_depth'] = 100

clf = lgb.train(params, D_train, 10000)
```

위 알고리즘에서 learning_rate는 학습률을 의미하며 디폴트 값인 0.003으로 정의하였다. boosting_type은 앙상블 모델에서 실행하고자 하는 부스팅 알고리즘 타입을 의미하며, 역시 디폴트 값인 gbdt로 설정하였다. objective와 metric 속성 값은 이진 분류 문제를 풀 수 있도록

각각 binary와 binary_logloss로 정의하였다. 나머지 3개의 파라미터는 결정 트리의 형태를 정의하는데 이용된다. num_leaves, min_data, max_depth는 차례로 전체 트리에서 리프 노드의 수, 리프 노드가 가지는 최소한의 레코드 수, 트리의 최대 깊이를 나타낸다. 이들 값은 데이터 특성에 따라 가변적으로 조정 가능하다.

3. Experimental Results

실험에서는 Table 1에서 제시된 학습 데이터를 대상으로 이전 절에서 설명한 DNN과 LGBM 모델의 예측 성능을 비교하였다. 실험은 Intel Xeon E5-2609 1.70GHz CPU와 MSI GeForce RTX-4090 GPU, 32GB 메모리가 장착된 HP 서버에서 수행되었다. 학습 알고리즘은 Tensor Flow 2.10.1을 이용하여 구현되었으며, GPU 가속 라이브러리로 CUDA 11.8이 이용되었다.

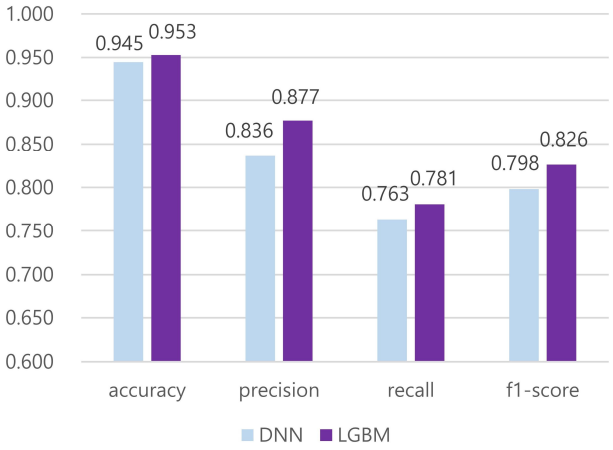

Fig. 4. Performance of dropout prediction of DNN and LGBM in terms of accuracy, precision, recall, and f1-score

먼저 DNN과 LGBM의 예측 성능을 비교하였다. Fig. 4는 두 모델의 예측 성능 결과를 정확도, 정밀도, 재현율, F1-스코어 측면에서 보여준다. 정확도 측면에서 두 알고리즘은 각각 0.945와 0.953의 성능을 보였으며, 단순히 중도 탈락하는 것으로 예측한 결과와 비교했을 때 유사하거나 조금 높은 성능을 제공하는 것으로 조사되었다. 두 알고리즘의 정밀도는 각각 0.836과 0.877이었으며, 재현율은 0.763과 0.781로 확인되었다. 두 알고리즘 모두에서 정밀도에 비해 재현율이 낮았으며, 재현율이 알고리즘의 F1-스코어에 더욱 큰 영향을 주는 것으로 확인되었다. DNN과 LGBM의 F1-스코어는 각각 0.798과 0.826으로 조사되었으며, LGBM이 DNN에 비해 2.5% 나은 예측 성능을 제공하는 것으로 확인되었다.

Table 3. Performance comparison of the proposed model and the existing models discussed in [13, 14, 15]

Measure	Existing models			Proposed model	
	Ref No.	Score	Algorithm	Score	Algorithm
Accuracy	[13]	**0.8**	DNN	**0.953**	LGBM
Precision	[14]	**0.739**	Ridge Regression	**0.877**	LGBM
F1-score	[15]	**0.8**	DNN	**0.826**	LGBM

다음으로 제안한 방법과 기존 연구에서 제시한 모델의 성능을 비교하였다. II장에서 언급한 바와 같이, 비교 대상은 4년제 대학교 학생들의 중도 탈락 예측 문제를 다룬 [13, 14, 15]로 설정하였다. Table 3은 기존 모델과 제안한 LGBM 모델의 성능을 비교하였다. [13]에서 가장 성능이 우수한 모델은 DNN 모델이었으며, 0.8의 정확도를 제공하였다. [14]에서는 릿지 회귀 모델이 가장 우수하였으며, 0.739의 정밀도를 제공하였다. [15]에서는 DNN 모델의 성능이 가장 우수하였으며, 0.8의 F1-스코어를 제공하였다. 제안한 LGBM 모델의 정확도와 정밀도, F1-스코어는 각각 0.953, 0.877, 0.826으로, 기존 모델과 비교했을 때 모든 측면에서 우수한 성능을 제공하였다.

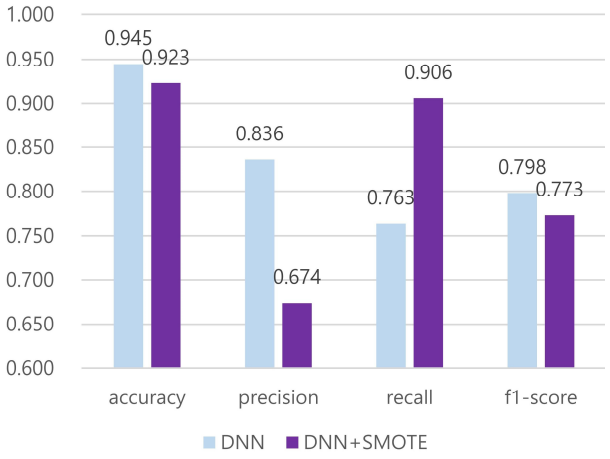

Fig. 5. Performance of dropout prediction of DNN and DNN+SMOTE in terms of accuracy, precision, recall, and f1-score

마지막으로 데이터 불균형이 모델의 성능에 미치는 영향을 파악하기 위해, 오버샘플링 전후의 성능 차이를 비교하였다. Fig. 5는 DNN 모델을 대상으로 SMOTE 적용 전과 후의 예측 성능과 관련하여 정확도, 정밀도, 재현율, F1-스코어 측면에서 비교하였다. 실험 결과, 모델의 F1-스코어는 0.798에서 SMOTE 적용 후 0.773으로 낮아졌으며, 오버샘플링을 적용한 후 예측 성능이 더욱 떨어지는 것으로 조사되었다. 한 가지 주목할 점은 SMOTE 적용 전

에는 정밀도 값이 재현율에 비해 높았던 반면, 적용 후에는 재현율 값이 더욱 높다는 점이다. 이는 학습 시 중도 탈락 데이터를 포함한 P 클래스로의 과적합이 발생했음을 의미한다. 그리고 정밀도가 낮아졌다는 의미는 오버샘플링된 P 클래스의 데이터가 N 클래스의 데이터 영역과 겹치는 부분이 커져 노이즈가 많아졌음을 의미한다.

결과적으로 중도 탈락 예측 문제에서 SMOTE를 이용한 단순한 오버샘플링 기법만으로는 예측 성능이 향상될 수 없음을 확인하였다. 성능 향상을 위해서는 이상치 탐지 및 클래스 간의 경계선(Border-line) 등을 함께 고려한, 보다 세밀한 오버샘플링 기법이 필요함을 확인할 수 있었다.

참고로 LGBM을 대상으로 한 SMOTE 적용 후의 성능 변화는 확인할 수 없었다. LGBM 설치 후 SMOTE를 실행할 경우 관련 패키지에서 오류가 발생하였다. 관련 해결책에 대해서는 지속적으로 모니터할 예정이다.

V. Conclusion and Future Work

본 논문에서는 서울 소재 중소규모 4년제 대학인 A 대학의 학사 정보 시스템에서 2010년부터 2022년까지 20,050명의 학생을 대상으로 수집된 학적 및 성적 데이터로부터 대학생들의 중도 탈락 여부를 예측하기 위한 지도 학습 모델을 구현하였다. 원본 데이터는 중복을 피하기 위해 여러 테이블로 분산된 형태로 저장되어 있었으며, 학습을 위해 하나의 테이블로 통합하였다. 그리고 상관관계 분석을 통해 학습에 밀접한 영향을 주는 8개의 속성을 선별하여 학습에 이용하였다. 학습 알고리즘으로는 기본적인 신경망 알고리즘인 DNN과 함께, 적은 비용으로 높은 분류 정확도를 제공하는 것으로 알려진 LGBM을 이용하여 모델을 구현하였다. 실험 결과, DNN과 LGBM 모델의 F1-스코어는 각각 0.798과 0.826이었으며, LGBM이 DNN에 비해 2.5% 나은 예측 성능을 제공하는 것으로 조사되었다. 특히 LGBM 모델의 예측 성능은 기존 연구에서 제시한 모델들의 성능에 비해 우수한 것으로 확인되었다.

제안 방법에서는 두 알고리즘의 성능 비교에 있어 중도 탈락 데이터가 가진 불균형이 예측 성능에 어떤 영향을 미치는지 함께 확인하고자 하였다. 이를 위해 DNN 알고리즘을 대상으로 SMOTE를 이용한 오버샘플링 적용 전과 후의 예측 성능을 비교하였다. 실험 결과 SMOTE 적용 후의 성능이 오히려 낮아졌으며, 단순한 오버샘플링 기법 적용만으로는 성능 향상이 어려움을 확인하였다.

향후에는 이상치 탐지 및 클래스 간 경계 등을 고려한 오버샘플링 기법을 중도 탈락 예측에 적용하는 방안에 대해 연구할 계획이다. 그리고 제안한 모델의 예측 성능을 높이기 위해 입학 유형별이나 학년별로 데이터를 분리하여 각각에 특화된 예측 모델을 만들고 이들을 앙상블하기 위한 방법에 대해서도 연구를 지속할 계획이다.

REFERENCES

[1] J. H. Kim, "The hierarchical relationship between individual, college, social variable, and dropout intention of college students," The Journal of Educational Research, Vol. 30, No. 2, pp. 249-266, 2011.

[2] H. B. Lee, "Analysis of main factors for college students underachievement," Humanities and Social Sciences 21, Vol. 7, No. 3, pp. 653-672, 2016. DOI: 10.22143/hss21.7.3.34

[3] C. Park, "Development of prediction model to improve dropout of cyber university," Journal of the Korea Acedemia-Industrial Cooperation Society, Vol. 21, No. 7, pp. 380-390, 2020.

[4] G. B. Moon, J. W. Kim, and J. S. Lee, "Early prediction model of student performance based on deep neural network using massive LMS log data," The Journal of the Korea Contents Association, Vol. 21, No. 10, pp. 1-10, 2021.

[5] J. Schmidhuber, "Deep learning in neural networks: an overview," Neural Networks, vol. 61, pp. 85-117, 2015. DOI: 10.1016/j.neunet.2014.09.003

[6] G. Ke et al. "LightGBM: A highly efficient gradient boosting decision tree," Advances in Neural Information Processing Systems, vol. 30, 2017.

[7] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321-357, 2002. DOI: 10.1613/jair.953

[8] E. H. Seo and E. Y. Kim, "Exploring the predictors of academic probation in college: focusing on variables of student engagement," Journal of the Korea Contents Association, vol.21, no.7, pp.469-476, 2021.

[9] H. J. Lee, "Study for prediction system of learning achievements of cyber university students using deep learning based on Autoencoder," Journal of Digital Contents Society, vol.19, no.6, pp.1115-1121, 2018. DOI: 10.9728/ dcs.2018.19.6.1115

[10] J. Y. Yang, K. H. Moon, and S. H, Park, "Extracting characteristics of underachievers learning using artificial intelligence and researching a prediction model," Journal of the Korea Institute of Information and Communication Engineering, vol.26, no.4, pp.510-518, 2022.

[11] C. H. Hwang, "Improvement of early prediction performance of under-performing students using anomaly data," Journal of the

Korea Institute of Information and Communication Engineering, vol.26, no.11, pp.1608-1614, 2022.

[12] ROC, Wikipedia: https://en.wikipedia.org/wiki/Receiver_operating_characteristic

[13] J. M. Kim and D. C. Lee, "A study on management of student retention rate using association rule mining," Journal of the Korea Industrial Information Systems Research, vol.23, no.6. pp.67-77, 2018.

[14] S. H. Jeong, "A study on the development of university students dropout prediction model using classification technique," Journal of Convergence Consilience, vol.5, no.2, pp.174-185, 2022. DOI: 10.33090/sfcc.5.2.11

[15] J. Lee, D. Kim, and J. Kil, "A study on the prediction model for student dropout," In Proc, of the Annual Conference of KIPS 2018, vol.25, no.1, pp.37-40, 2018.

[16] H. S. Park and S. J . Yoo, "Early Dropout Prediction in Online Learning of University using Machine Learning," International Journal on Informatics Visualization, vol.5, no.4, pp.347-353, 2021. DOI: 10.30630/joiv.5.4.732

[17] W. Xing and D. Du, "Dropout prediction in MOOCs: Using deep learning for personalized intervention," Journal of Educational Computing Research, vol.57, no.3, pp.547-570, 2019. DOI: 10.1177/0735633118757015

[18] S. Lee and J. Y. Chung, "The Machine Learning-Based Dropout Early Warning System for Improving the Performance of Dropout Prediction," Applied Sciences, vol.9, no.15, 3093, 2019. DOI: 10.3390/app9153093

[19] Exponential smoothing, Wikipedia: https://en.wikipedia.org/wiki/Exponential_smoothing

[20] Google Keras: https://www.tensorflow.org/guide/keras

Authors

Hyeon Gyu Kim received the B.S. and M.S. degrees in Computer Science from University of Ulsan, and Ph.D. degree in Computer Science from Korea Advanced Institute of Science and Technology, Korea, in 1997, 2000, and 2010, respectively. Dr. Kim joined the faculty of the Division of Computer Science and Engineering at Sahmyook University, Seoul, Korea, in 2012. He is currently an Associate Professor in the Division of Computer Science and Engineering, Sahmyook University. He is interested in artificial intelligence and big data processing.